

Description

METHOD AND RELATED APPARATUS FOR CONTROLLING DATA TRANSMISSION IN A MEMORY

BACKGROUND OF INVENTION

[0001] 1. Field of the Invention

[0002] The present invention relates to a method and related apparatus for controlling data transmission in a memory, and more particularly, to a method and related apparatus for controlling data transmission in a memory without the involvement of a CPU.

[0003] 2. Description of the Prior Art

[0004] Please refer to Fig.1. Fig.1 is a schematic diagram of a computer system 10. As shown in Fig.1, the computer system 10 includes a CPU 12, a north bridge circuit 14, a south bridge circuit 16, a display controller 18, a display 19, a memory 20, a hard disc 22, and an input device 24. The memory 20 includes a plurality of memory units 26

arranged in arrays, i.e. each memory unit 26 corresponds to a column address and a row address. When the computer system 10 operates, the CPU 12 loads data stored in the memory 20 to a register 28 therein. The data held in the register 28 is then processed, and sent back to the memory 20. The accessing operation of the data stored in the memory 20 is processed via a memory controller 30 of the north bridge circuit 14. The memory controller 30 includes an address register 32 and a data register 34, where the address register 32 is for storing memory addresses and the data register 34 is for storing data to be written in the memory 20 and data retrieved from the memory 20. For example, when the CPU 12 executes an instruction and therefore has to move a data bit D stored in a memory unit 26a to a memory unit 26b, the CPU 12 sends an address data ADDRESSa (a physical memory address) corresponding to the memory unit 26a to the address register 32. The memory controller 30 can therefore retrieve the data bit D stored in the memory unit 26a in accordance with the address data ADDRESSa, and store the data bit D in the data register 34. The memory controller 30 then delivers the data bit D to the register 28. Since the purpose of this instruction is to move the data

bit D, no logic operations toward the data bit D are necessary. The CPU 12 only outputs an address data ADDRESSb to the address register 32 and delivers the data bit D held in the register 28 to the data register 34 so that the memory controller 30 writes the data bit D in the memory unit 26b in accordance with the address data ADDRESSb.

[0005] In the process of moving the data bit D stored in the memory unit 26a to the memory unit 26b, the CPU 12 does not need to execute any operations with respect to the data bit D. However, it takes a plurality of clock cycles for the CPU 12 to move the data bit D held in the data register 34 to the register 28 and to deliver the data bit D held in the register 28 to the data register 34. As a result, the load on the CPU 12 is increased. In addition, the transmission of the data bit D consumes the bandwidth of the front-side bus (FSB) between the CPU 12 and the north bridge circuit 14.

SUMMARY OF INVENTION

[0006] It is therefore a primary objective of the present invention to provide a method and related apparatus for controlling data transmission within a memory to solve the above problems.

[0007] According to the claimed invention, a method for control-

ling data transmission within a memory of a computer system is disclosed. The computer system comprises a processor, and a memory controller connected to the processor and the memory. The method comprises delivering a plurality of data located in a plurality of first memory addresses of the memory to the memory controller, and the memory controller directly storing the data in a plurality of second memory addresses of the memory instead of transmitting the plurality of data to the processor.

[0008] The present invention further provides a computer system comprising a processor for controlling operations of the computer system, a memory including a plurality of first memory addresses and second memory addresses, and a memory controller electrically connected to the processor and the memory. The memory controller has an internal data transmission controller for retrieving a plurality of data according to the first memory addresses, and directly storing the plurality of data in the second memory addresses instead of transmitting the plurality of data to the processor.

[0009] The internal data transmission controller of the computer system is engaged in processing the transmission of a data bit stored in the memory from one memory address

to another. Consequently, the transmission is processed without the involvement of the CPU of the computer system.

[0010] These and other objectives of the present invention will no doubt become obvious to those of ordinary skill in the art after having read the following detailed description of the preferred embodiment that is illustrated in the various figures and drawings.

BRIEF DESCRIPTION OF DRAWINGS

[0011] Fig.1 is a schematic diagram of a computer system.

[0012] Fig.2 is a schematic diagram of a computer system in a first embodiment of the present invention.

[0013] Fig.3 is a schematic diagram of a memory address table of a memory.

[0014] Fig.4 is a schematic diagram of a computer system in a second embodiment of the present invention.

DETAILED DESCRIPTION

[0015] Please refer to Fig.2. Fig.2 is a schematic diagram of a computer system 80 in a first embodiment of the present invention. As shown in Fig.2, the computer system 80 includes a CPU 82, a north bridge circuit 84, a south bridge circuit 86, a display controller 88, a memory 90, an input

device 92, a hard disc 94, and a display 96. The north bridge circuit 84 has a memory controller 98, which includes not only an address register 102 and a data register 104 as the conventional memory controller, but also an internal data transmission controller 100 engaged in processing the transmission of data within the memory 90. With the internal data transmission controller 100, data of the memory 90 can be moved from a memory unit 106a to a memory unit 106b inside the north bridge circuit 84. The accessing operations of the CPU 82 are no longer required.

[0016] The internal data transmission controller 100 is engaged in processing the transmission of a data bit D from the memory unit 106a to the memory unit 106b. In the beginning of the transmission, a memory address ADDRESSa corresponding to the memory unit 106a is held in the address register 102. Then the internal data transmission controller 100 reads the memory unit 106a according to the memory address ADDRESSa, and stores the data bit D in the data register 104. Following that, a memory address ADDRESSb corresponding to the memory unit 106b is delivered to the address register 102, and the internal data transmission controller 100 writes the data bit D held

in the data register 104 to the memory unit 106b according to the memory address ADDRESSb. It is worth noting that the data bit D is not delivered to the CPU 82 in the course of the transmission. As a result, the loading of CPU 82 is reduced, and the bandwidth of the FSB between the CPU 82 and the north bridge circuit 84 is not consumed.

[0017] In addition, if the data of the memory 90 to be transmitted includes a plurality of data bits, the memory controller 98 uses physical memory addresses (e.g. a memory address table) to access the memory units 106 of the memory 90. Please refer to Fig.3. Fig.3 is a schematic diagram of a memory address table 107 of the memory 90. As shown in Fig.3, the memory address table 107 includes three kinds of fields where fields 108a, 108b, and 108n record the physical memory addresses, fields 110a, 110b, and 110n record flags which represent whether the data is an end portion (end of file, EOF), and fields 112a, 112b, 112n designate a bit length of each physical memory address recorded in fields 108.

[0018] When a program needs to process a data transmission in the memory 90, the program requests the operating system of the computer system 80 to obtain the physical memory addresses corresponding to the data stored in

the memory 90, and generates the memory address table 107 stored in a predetermined block of the memory 90. Then the program outputs an instruction to command the internal data transmission controller 110 to retrieve data correctly according to the memory address table 107. Accordingly, the internal data transmission controller 100 reads the memory address ADDRESS_a recorded in field 108a, retrieves a plurality of data bits from the memory address ADDRESS_a in accordance with a bit length LENGTH_a recorded in field 112a, and consecutively writes the data bits to the address register 102. Since the flag recorded in field 110a is "0", i.e. the data is not an end portion, the internal data transmission controller 100 then reads the memory address ADDRESS_b recorded in field 108b, and retrieves a plurality of data bits from the memory address ADDRESS_b in accordance with a bit length LENGTH_b recorded in field 112b. Similarly, since the flag recorded in field 110b is "0", the internal data transmission controller 100 keeps on repeating the same action. The internal data transmission controller 100 will read the memory address ADDRESS_n, retrieve a plurality of data bits according to a bit length LENGTH_n recorded in field 112n, and stop since the flag recorded in field 110n is "1",

i.e. end of file (EOF). In a similar manner, if the memory controller 98 needs to write data to the memory 90, the operating system will generate a memory address table 107 as shown in Fig.3 for recording the physical memory addresses. Accordingly, the internal data transmission controller 100 can write data held in the data register 104 to the memory units 106 of the memory 90 according to the memory address table 107.

[0019] It is noted that if the data stored in the memory 90 correspond to a plurality of physical memory addresses 106a which are discontinuous, the memory address table 107 is required to read the data. Similarly, when the physical memory addresses 106b to where the data will be moved are discontinuous, the memory address table 107 is also required. However, if the data stored in the memory 90 correspond to a plurality of physical memory addresses that are continuous, the operating system only has to provide a source memory address, a bit length, and a target memory addresses so that the internal data transmission controller 100 can consecutively read the data bits from the source memory address according to the bit length, and write the data bits to the target memory address. Certainly, there may be more than one target ad-

dress, and in such case the memory address table 107 is also required to write the data bits to different target memory addresses.

[0020] Please refer to Fig.4. Fig.4 is a schematic diagram of a computer system 120 in a second embodiment of the present invention. As shown in Fig.4, the computer system 120 includes a CPU 122, a north bridge circuit 124, a south bridge circuit 126, a display 128, a memory 130, an input device 132, and a hard disc 134. The north bridge circuit 124 includes a memory controller 136, and a display controller 138. The memory controller 136 further includes an internal data transmission controller 140, an address register 142, and a data register 144. The memory 130 is divided into a system memory 148 and a display memory 150 both comprising a plurality of memory units 152 arranged in arrays. The computer system 120 adopts a unified memory architecture (UMA), and thus the CPU 122 and the display controller 138 share the memory 130 for accessing data. In other words, the CPU 122 uses the system memory 148 while the display controller 138 uses the display memory 150. Note that the components having the same terminology in Fig.4 and Fig.2 have the same function, and thus redundant descriptions are not

given herein. In this embodiment, the memory controller 140 transmits a data bit D from a memory unit 152a to a memory unit 152c of the display memory 150, or transmits a data bit D from a memory unit 152c of the display memory 150 to a memory unit 152a. The transmission is carried out by the internal data transmission controller 140, instead of by the CPU 122. Consequently, the load on the CPU 122 is reduced, and the bandwidth of the FSB between the CPU 122 and the north bridge circuit 124 is not consumed.

[0021] Similar to the first embodiment of the present invention, the internal data transmission controller 140 can transmit data whether the physical memory addresses are continuous or not. If the physical memory addresses are discontinuous, the internal data transmission controller 140 transmits data in accordance with the memory address table 107 (shown in Fig.3). If the physical memory addresses are continuous, the internal data transmission controller 140 only needs a source memory address for designating the start address, a bit length, and a target memory address for designating where in the memory 130 the data is to be moved to execute the transmission.

[0022] The memory controller of the computer system includes

an internal data transmission controller engaged in transmitting data within the memory. When data bits stored in a memory address need to be transmitted to another memory address in the memory, the internal data transmission controller reads the data bits, stores the data bits in the data register, and stores the data bits in another memory address of the memory. It is clear that the transmission of the data bits is completely executed by the internal data transmission controller, without the involvement of the CPU. As a result, the loading of the CPU is reduced. In addition, the computer system of the present invention is more efficient since the bandwidth of the FSB between the CPU and the north bridge circuit is not consumed in the course of the transmission.

[0023] Those skilled in the art will readily appreciate that numerous modifications and alterations of the device may be made without departing from the scope of the present invention. Accordingly, the above disclosure should be construed as limited only by the metes and bounds of the appended claims.